Boosting StoRM Convergence with Metric Guidance and Non-uniform State-Sampling for Optimal Dereverberation

Chandra Mohan Sharma[1,2], *Arnab Kumar Roy*[1], *Anupam Mandal*[2], *Prasanta Kumar Ghosh*[1], *Prasanna Kumar KR*[2]

[1]Department of Electrical Engineering, Indian Institute of Science (IISc), India
[2]Center for Artificial Intelligence and Robotics (CAIR), DRDO, India

chandrams@iisc.ac.in, arnabroy.cair@gmail.com, amandal.cair@gov.in, prasantg@iisc.ac.in, prasanna.cair@gov.in

Abstract

This paper proposes a novel approach to address late reverberation, which degrades speech intelligibility by convolving clean speech with room impulse response. Our method combines metric-guided training and non-uniform state sampling within the Stochastic Regeneration Model (StoRM) diffusion architecture, enabling better diffusion variability modeling while maintaining computational efficiency. Key metrics such as STFT loss, spectral convergence loss, Mel Frequency Cepstral Coefficient (MFCC) loss and log-magnitude loss guide the regeneration process, improving convergence by reducing training epochs by $\sim \mathbf{19.6}\%$ with slight improvements in dereverberation. Meanwhile, the non-linear state sampling approach enhances training convergence by $\sim \mathbf{27.2}\%$ with practically similar perceptual performance. We evaluate the impact of these modifications on automatic speech recognition and clean speech distortion relative to the baseline, demonstrating optimal speech-quality-aware performance.

Index Terms: Dereverberation, Metric-Guided StoRM, Non-Uniform State Sampling, PESQ Loss, STFT Loss, MFCC Loss, Log-magnitude Loss, StoRM, SDE, WER, CER, ASR.

1. Introduction

Late reverberation significantly degrades speech intelligibility, introducing temporal and frequency variability that complicates dereverberation. Unlike additive noise, which can be modelled as $y(t) = x(t) + n(t)$, where $y(t)$ is the noisy signal, $x(t)$ the clean signal, and $n(t)$ is the noise, reverberation is modelled as $y(t) = x(t) * h(t)$, where $h(t)$ being the room impulse response (rir). In addition to convoluted nature, reverberation is a non-stationary phenomenon, affected by environmental changes and positional variations [1] [2]. Classical methods such as inverse filtering [3], spectral subtraction [4], and adaptive filtering [5] have limitations in handling the complexity of reverberation. Modern deep learning approaches, including predictive models like Denoising Autoencoders (DAEs) [6], U-Nets [7], and generative models like Variational Autoencoders (VAEs) [8], Generative Adversarial Networks (GANs) [9] [10] and High Fidelity GAN [11] offer improved performance. However, most of the deep learning methods tend to introduce additional distortions when applied to already clean speech, thereby limiting their optimal performance. In particular, generative models often suffer from slow convergence and mode collapse, further hindering their practical effectiveness.

This paper investigates the application of diffusion-based generative models [1, 12–14] for effective and optimal dereverberation tasks, focusing on the diffusion model implemented within the StoRM [13] framework. The StoRM [13] framework has shown superior performance compared to other leading architectures. Although these models provide a more stable learning process, they often encounter challenges with slow convergence [15] and they also introduce distortion in speech features while processing clean speech. To address these issues, we propose two techniques: metric based regularizer guidance within the loss function and non-uniform state sampling. These methods significantly accelerate convergence while training and achieve slightly improved or similar perceptual and Automatic Speech Recognition(ASR) performance compared to baseline. These regularizers also reduce distortion introduced by the baseline StoRM [13] when processing clean speech by selectively enhancing reverb signal from the mixture of clean and reverb signal.

2. Brief Description of StoRM [13]

In the StoRM [13] framework, a predictive model, D_θ, is combined with a generative diffusion model, G_ϕ, to facilitate a stochastic regeneration process. In this approach, the predictive model D_θ first transforms the observed data $\mathbf{y}$ into the posterior mean, $\mathbf{y} \rightarrow \mathbb{E}[\mathbf{x} \mid \mathbf{y}]$. Then, the generative model refines this posterior mean through iterative steps, progressively guiding the sample towards the high-probability regions of the posterior $p(\mathbf{x} \mid \mathbf{y})$ space. The StoRM framework operates through both a forward and a reverse process, as illustrated in Figure 1.

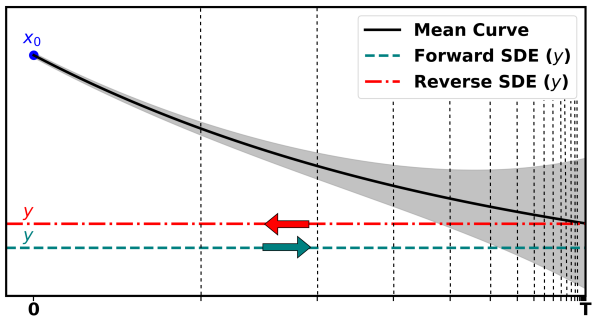

Figure 1: *Depiction of the forward and reverse SDE processes within the model, illustrating higher variance near T compared to 0 motivating for T-biased sampling.*

In the forward process $\{\mathbf{x}_\tau\}_{\tau=0}^{T} \mid \tau \sim \text{Uniform}(0, T]$, a clean sample $\mathbf{x}_0$ is progressively corrupted by adding noise, resulting in a noisy sample $\mathbf{x}_T$, hence the current state is

$$\mathbf{x}_\tau = \mu(\mathbf{x}_0, \mathbf{y}, \tau) + \sigma(\tau)\mathbf{z}, \quad \mathbf{z} \sim \mathcal{N}(\mathbf{z}; 0, I),$$

with $\mu(\mathbf{x}_0, \mathbf{y}, \tau)$ is mean & $\sigma^2(\tau)$ being the variance at τ. The reverse diffusion process, on the other hand, begins with the

corrupted sample $\mathbf{x}_T$ and progressively refines it, following the reverse trajectory through time steps from T to 0, as governed by a discretized reverse stochastic differential equation (SDE) [16]. This iterative process ultimately restores the clean speech sample $\mathbf{x}_0$.

In the StoRM [13] framework, the generative process is guided by the posterior distribution $p(\mathbf{x} \mid \mathbf{y})$, enabling iterative refinement of the noisy or reverberated signal towards the clean speech. As a result, the corrupted input $\mathbf{y}$ is transformed into an estimated clean output $\hat{\mathbf{x}}$, as illustrated in Figure 2.

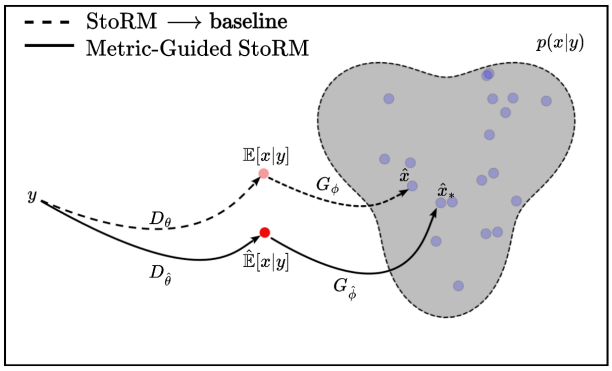

Figure 2: *Illustration of the metric-guided StoRM framework, showing generation of a metric-guided posterior mean* $[\mathbf{y} \rightarrow [D_{\hat{\theta}}(y)] \rightarrow \hat{\mathbb{E}}[\mathbf{x} \mid \mathbf{y}]]$*, followed by the estimation of a cleaner sample* $[G_{\hat{\phi}}(\hat{\mathbb{E}}[\mathbf{x} \mid \mathbf{y}]) \rightarrow \hat{\mathbf{x}}_\star]$*. Grey area is posterior space & the dotted path represents the baseline StoRM approach.*

3. Methodology

In StoRM [13] diffusion model the score function $\nabla_{\mathbf{x}_\tau} \log p_\tau(\mathbf{x}_\tau)$ is the gradient of the log probability density of the noisy data $\mathbf{x}_\tau$ at diffusion time τ. The score function $\nabla_{\mathbf{x}_\tau} \log p_\tau(\mathbf{x}_\tau)$ is not tractable and therefore it is estimated by a deep neural network (DNN) s_ϕ, referred to as the score model. The complete loss function of StoRM [13] combines the denoising score matching objective and the L_2 regression loss of the predictive model, as shown in Eq. 1

$$L(\phi,\theta) = \mathbb{E}_{\tau,(\mathbf{x},\mathbf{y}),\mathbf{z}}\left[\left\| s_\phi(\mathbf{x}_\tau,[\mathbf{y},D_\theta(\mathbf{y})],\tau) + \frac{\mathbf{z}}{\sigma(\tau)}\right\|_2^2\right] + \mathbb{E}_{(\mathbf{x},\mathbf{y})}\left[\|\mathbf{x} - D_\theta(\mathbf{y})\|_2^2\right] \tag{1}$$

To guide the regeneration process and to reduce the distortion of speech features during dereverberation, we incorporated various regularizers into the original loss function, which comprises of Short Time Fourier Transform (STFT) loss [7], log-magnitude (MAG) loss [7], Spectral convergence (SC) loss [7] and MFCC loss as in Eq.2-5. These modifications result in accelerated convergence, while obtaining the optimal dereverberation performance by minimizing spectral feature losses.

Following the methods in [7], we integrated the regularizers mentioned above as shown in Eq.6-8 to create metric-guided variants of StoRM. These regularizers assist in guiding both the predictive and diffusion processes during training, as illustrated in Figure 2. We evaluated different combinations of regularizers, and the most effective ones, along with the modified loss function, are detailed in Eq.2-8.

$$L(\text{STFT}) = \frac{1}{N}\frac{\||\text{STFT}(x)| - |\text{STFT}(\hat{x})|\|_F}{\||\text{STFT}(x)|\|_F} \tag{2}$$

$$L(\text{MFCC}) = \frac{1}{N}\frac{\||\text{MFCC}(x)| - |\text{MFCC}(\hat{x})|\|_F}{\||\text{MFCC}(x)|\|_F} \tag{3}$$

$$L(\text{MAG}) = \frac{1}{N}\|\log|\text{STFT}(x)| - \log|\text{STFT}(\hat{x})|\|_1 \tag{4}$$

$$L(\text{SC}) = \frac{1}{M}\sum_{m=1}^{M}\left(\frac{1}{N}\frac{\||\text{STFT}_m(x)| - |\text{STFT}_m(\hat{x})|\|_F}{\||\text{STFT}_m(x)|\|_F}\right) \tag{5}$$

$$L(\phi,\theta,\text{STFT}) = L(\phi,\theta) + L^2(\text{STFT}) \tag{6}$$

$$L(\phi,\theta,\text{STFT},\text{MFCC}) = L(\phi,\theta) + L(\text{STFT}) + L(\text{MFCC}) \tag{7}$$

$$L(\phi,\theta,\text{SC}^2,\text{MAG}^2) = L(\phi,\theta) + L^2(\text{SC}) + L^2(\text{MAG}) \tag{8}$$

Here, x and $\hat{x}$ are the clean and estimated speech signals, respectively, N denotes the number of time frames in the speech segments and M denotes the number of STFT losses with different STFT resolutions i.e. multiple FFT points as in [7].

As shown in Figure 1, the diffusion process exhibits a high variance toward T, containing richer information. To capture these variations more effectively, we conducted an additional experiment in which an exponential transformation was applied to uniformly distributed samples of τ. This modification was performed without any addition of regularizer. As the state transitions from 0 to T, the diffusion variance increases, while the posterior mean decays exponentially. This nonuniform sampling, with a higher density near T, accelerates convergence while maintaining comparable performance, as highlighted in Table 1.

In StoRM [13] training, the sample diffusion time is uniformly selected as:

$$\tau \sim U(0,T] \Longrightarrow L(\phi,\theta) \tag{9}$$

To focus more samples around T, we apply the following transformation to uniformly distributed samples:

$$\tau_{nu} = T\left(1 - e^{-\lambda\tau}\right) \Longrightarrow L(\phi,\theta,\tau_{\text{non-uniform}} \rightarrow T) \tag{10}$$

Conversely, to concentrate more samples around 0, we use the following transformation:

$$\overline{\tau}_{nu} = Te^{-\lambda\tau} \Longrightarrow L(\phi,\theta,\tau_{\text{non-uniform}} \rightarrow 0) \tag{11}$$

where $\lambda = 5$ is a hyperparameter that controls the concentration of samples by increasing the exponent value, leading to a higher density of samples near extreme points. As exponent increases in Eq. 10, τ_{nu} increases, resulting in fewer samples farther from T. In contrast, according to Eq. 11 as exponent increases, $\overline{\tau}_{nu}$ decreases, causing the samples to concentrate closer to 0.

Experimental results as in Fig. 4 confirm that Eq. 10 effectively supports our hypothesis that T-biased sampling captures more information. Hence, incorporating Eq.6-11, the original StoRM [13] training algorithm is modified as Algorithm 1.

Table 1: *Evaluation of dereverberation on the test dataset (2000) across different models, with mean ($\bar{\mu}$) and standard deviation ($\bar{\sigma}$) for each score. SGMSE+M, StoRM & our proposed variants use $N = 30$ steps. '*' $\Rightarrow$ proposed StoRM Variant. Best results are in* ***bold****.*

Methods (*StoRM variants $\leftrightarrow$ loss function*)	**Evaluation Metrics**					
	PESQ	**CSIG**	**CBAK**	**COVL**	**STOI**	**Epochs**
Reverb (*unprocessed*)	1.6136 ± 0.0568	2.9701 ± 0.3439	1.7380 ± 0.0973	2.2665 ± 0.1811	0.6631 ± 0.0092	–
$L(\phi, \theta) \rightarrow$ *baseline*	3.1601 ± 0.1152	4.4790 ± 0.6475	3.1569 ± 0.0553	3.8363 ± 0.4774	0.9378 ± 0.0010	504
$L(\phi, \theta, \text{STFT})$*	3.1783 ± 0.1147	4.4795 ± 0.6494	$\mathbf{3.1602 \pm 0.0580}$	3.8366 ± 0.4783	0.9385 ± 0.0009	424
$\boldsymbol{L(\phi, \theta, \mathbf{SC^2}, \mathbf{MAG^2})}$*	$\mathbf{3.2020 \pm 0.1142}$	$\mathbf{4.4878 \pm 0.6501}$	3.1205 ± 0.0512	$\mathbf{3.8505 \pm 0.4814}$	$\mathbf{0.9393 \pm 0.0009}$	479
$\boldsymbol{L(\phi, \theta, \tau_{\text{non-uniform}} \rightarrow T)}$*	3.1559 ± 0.1193	4.4547 ± 0.6475	3.1400 ± 0.0569	3.8121 ± 0.4774	0.9372 ± 0.0010	**367**
$L(\phi, \theta, \text{STFT}, \text{MFCC})$*	3.1947 ± 0.0041	4.4798 ± 0.0067	3.1377 ± 0.0023	3.8434 ± 0.0044	0.9387 ± 0.0001	405
SGMSE+M [13]	2.6443 ± 0.2034	3.9873 ± 0.6649	2.8064 ± 0.2343	3.3259 ± 0.4715	0.8886 ± 0.0032	Pre-trained
NCSN++M [13]	2.4461 ± 0.1839	3.4881 ± 0.5192	2.7178 ± 0.0837	2.9798 ± 0.3815	0.9079 ± 0.0018	Pre-trained
CM-GAN [9]	2.0599 ± 0.0931	3.3712 ± 0.3616	1.9335 ± 0.0481	2.6940 ± 0.2266	0.8138 ± 0.0023	263
TF-GAN (Voice-Fixer) [10]	1.4445 ± 0.0052	2.9021 ± 0.0201	1.7657 ± 0.0041	2.1782 ± 0.0098	0.7891 ± 0.0002	Pre-trained
MetricGAN-U [19]	1.6329 ± 0.0588	3.0252 ± 0.3461	1.8252 ± 0.1073	2.3085 ± 0.1839	0.6618 ± 0.0092	250

Algorithm 1 : Modified-StoRM Training

Input: Training set of pairs $(\mathbf{x}, \mathbf{y})$
Output: Trained parameters $\{\phi, \theta\}$

1: Uniform sample time $\tau \sim \mathcal{U}(\tau_\epsilon, T) \mid \tau_\epsilon = 0; T = 1$
2: T-biased diffusion time $\tau_{nu} = T(1 - e^{-\lambda\tau}) \mid \lambda = 5.0$
3: Sample noise signal $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$
4: Infer initial prediction $D_\theta(\mathbf{y})$
5: Generate perturbed state $\mathbf{x}_{\tau_{nu}} \leftarrow \mu(\mathbf{x}, D_\theta(\mathbf{y}), \tau_{nu}) + \sigma(\tau_{nu})\mathbf{z}$
6: Estimate score $\mathbf{s}_\phi(\mathbf{x}_{\tau_{nu}}, [\mathbf{y}, D_\theta(\mathbf{y})], \tau_{nu})$
7: Compute loss $L(\phi, \theta, f(\mu)) = \mathcal{J}^{\text{StoRM}}(\phi, \theta) + f(\mu)|$ $\tau, \mu \rightarrow$ metric; $L(\phi, \theta, \tau_{\text{non-uniform}}) = \mathcal{J}^{\text{StoRM}}(\phi, \theta)|$ T-biased diffusion time (τ_{nu})
8: Backpropagate loss $L(\phi, \theta, f(\mu))$ or $L(\phi, \theta, \tau_{\text{non-uniform}})$ to update $\{\phi, \theta\}$

4. Experiments

4.1. Dataset

In [13], the authors used the WSJ0+Reverb dataset, which was created by convolving WSJ0 speech data [17] with simulated rirs using the *pyroomacoustics* library-based utility, as detailed in the footnote[1]. For our experiments, we used LibriSpeech dataset [18], selecting 14,000 random utterances (10,000 for training, 2,000 for validation and 2,000 for testing). Each FLAC file was converted to a single-channel WAV file (16 kHz, 16-bit, linear PCM). For data generation, we used the same utility[1] as in the StoRM [13] framework to configure all parameters, including room dimensions $[5, 15] \times [5, 15] \times [2, 6]$ m, a minimum source-to-wall distance of 1 m, anechoic targets with an absorption coefficient of 0.99, and a noise floor of 50 dB. 14,000, rirs (T60: 0.4–2.5 s) thus generated were convolved with Librispeech clean dataset to create parallel corpora for our experiment. The StoRM code, data preparation utility and pre-trained model are downloaded from the online GitHub repository[1].

4.2. Experimental details

To train the dereverberation model in a supervised manner, we created a dataset of paired reverb and clean speech files, as detailed in Subsection 4.1. We selected a batch size of 4, with each batch comprising 2-second segments randomly sampled from pairs of clean and reverb speech signals. Following the method in [13], we applied a square-root Hann window (size 510, hop length 128) to convert each 2-second, 16 kHz speech frame into 256 frames, resulting in input tensors as [4,1,256,256].

Training used, a patience of 50 epochs, N=30 iterations for training, evaluation, and inference, and a total diffusion time of T=1. The baseline model was trained using a pre-trained model[1], extended to 504 epochs until the training patience. Further, the code was modified with loss functions and state sampling mechanisms as outlined in Section 3 and Algorithm 1. StoRM variants, which incorporate non-uniform state sampling and metric-guided learning (using STFT, MAG, MFCC & SC losses) were trained separately, with results and detailed comparisons provided in Table 1.

For evaluation 2000 file pairs from test dataset were processed using the trained model. The enhanced batch and its corresponding clean batch from the test dataset were used to compute performance metrics for comparison. Additionally, we evaluated our test dataset using several pre-trained models, including Voicefixer (TFGAN) [10], NCSN++M and SGMSE+M [13]. We have also trained the CMGAN [9], MetricGAN-U [19] models from SpeechBrain [20] [21] on our training dataset and evaluated their performance on test dataset. The results obtained were then compared with those from our modified StoRM models, as shown in Table 1. The performance of ASR on the dereverbed dataset, evaluated using the large Wav2Vec [22] model, showed the Word Error Rate (WER) & Character Error Rate (CER) comparable to the baseline as in Table 2. We also assessed the degradation introduced by StoRM variants by processing 2000 clean test files, the results are shown in Table 3. Code & inference data is available online. [2]

4.3. Evaluation metrics

To evaluate dereverberation performance, we employed several objective metrics [23] on dereverb test data. Speech quality was assessed using the Perceptual Evaluation of Speech Quality (PESQ) [24] in wide band mode, Short-Time Objective Intelligibility (STOI) [25], signal distortion (CSIG) [23], background noise (CBAK) [26] and overall quality (COVL) [27] scores. PESQ function used in our experiment is a Python package that wraps the ITU-T P.862 standard. These metrics offer a comprehensive evaluation of the dereverberation performance.

[1] StoRM code, data preparation utility and pre-trained model can be accessed at: `https://github.com/sp-uhh/storm`.

[2] `https://github.com/cmsiisc/StoRM-variants`

To evaluate accelerated convergence, we measured the number of epochs required for each variant of metric-guided StoRM to reach its saturation point at the patience limit during training. To evaluate perceptual and automatic recognition distortion introduced by the variants of StoRM, PESQ, WER & CER metrics are used. The results are shown in Table 3.

4.4. Results

Figure 3. illustrates that incorporating metric based regularizers into the loss function, as described in Section 3, accelerates convergence by requiring upto 19.6% fewer epochs and slightly improves dereverberation compared to the baseline, as shown in Table 1. As outlined in subsection 3, Figure 4 demonstrates that concentrating samples around T, instead of 0, captures more variability, resulting in a 27.2% faster convergence compared to the baseline, while maintaining comparable performance, as shown in Table 1. Conversely, sampling near 0 leads to diminished performance, as illustrated in Figure 4.

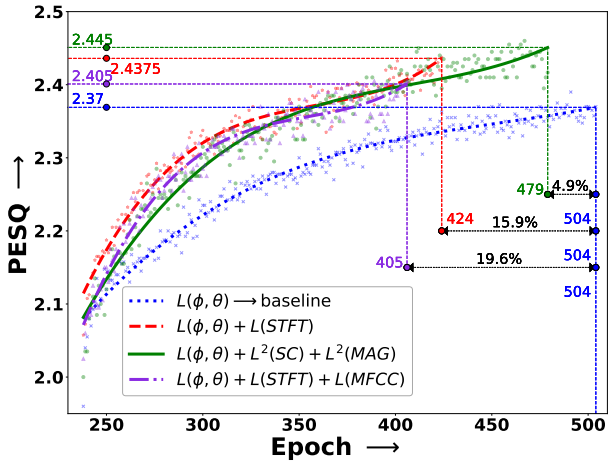

Figure 3: *PESQ vs epoch plot during validation, showing faster convergence (upto 19.6%) and improved performance of Metric-guided StoRM as in Section 3. w.r.t the baseline.*

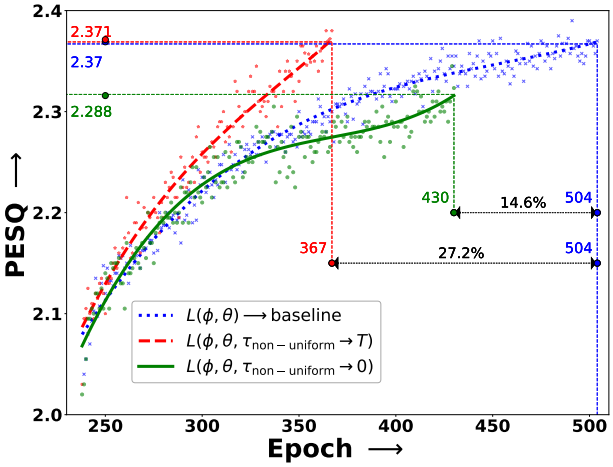

Figure 4: *PESQ vs. epoch plot showing that sampling more towards T (Eq. 10) outperforms uniform sampling (baseline) within 367^{th} epoch (27.2% faster). Conversely, sampling towards 0 (Eq. 11) results in degraded performance.*

The CMGAN [9] and MetricGAN-U [19] models were trained on our dataset, whereas for SGMSE+M [13], NCSN++M [13] and Voicefixer [10] we used pre-trained models. StoRM and its variants outperform competing models (Table 1), with mean ($\bar{\mu}$) and standard deviation ($\bar{\sigma}$) of scores from 2000 test pairs, demonstrating superior performance in handling long-tail reverberation (T60: 0.4–2.4 s). Tables 1 and 2 jointly show that all StoRM variants achieve faster convergence with nearly similar or improved perceptual and ASR performance. Paired *t*-tests showed statistically significant gains in each measured metric ($p < 0.05$). Table 3 further confirms that processing clean speech with regularizer based StoRM variants does not alter PESQ, WER or CER values, highlighting their optimal quality-aware enhancement, whereas the baseline StoRM reduces PESQ of clean speech from 4.5 to 4.24. Notably, a PESQ value greater than 4.5 is due to floating-point precision, reflecting near-perfect input pair matching as shown in Table 3.

Table 2: *ASR [22] CER & WER on 2000-dereverb test dataset remain nearly unchanged with StoRM variants vs. baseline*, ensuring optimal performance. Reverb data processing methods ↔ respective loss functions (StoRM variants).*

Processing methods	**CER(%)**	**WER(%)**
Clean (*unprocessed*)	0.69	2.81
Reverb (*unprocessed*)	11.97	26.01
$L(\phi,\theta) \rightarrow$ *baseline**	1.74	5.6
$L(\phi,\theta,\text{STFT})$	1.77	5.83
$L(\phi,\theta,\text{MFCC})$	1.77	5.73
$L(\phi,\theta,\text{STFT},\text{MFCC})$	1.72	5.51
$\boldsymbol{L(\phi,\theta,\text{SC}^2,\text{MAG}^2)}$	**1.70**	**5.11**
$L(\phi,\theta,\tau_{\text{non-uniform}} \rightarrow T)$	1.72	5.51

Table 3: *StoRM variants yield near-identity transformation while processing 2000-clean test dataset vs. baseline*, where PESQ drops (4.5 → 4.24), ensuring their optimal performance. PESQ > 4.5 stems from floating-point precision, indicating near-identical input pairs. Processing methods ↔ loss func.*

Processing methods	**PESQ**	**CER(%)**	**WER(%)**
Clean (*unprocessed*)	4.5	0.69	2.81
$L(\phi,\theta) \rightarrow$ *baseline**	4.24 ± 0.020	0.70	2.85
$L(\phi,\theta,\tau_{\text{non-uniform}} \rightarrow T)$	4.24 ± 0.018	0.71	2.83
$L(\phi,\theta,\text{STFT})$	4.51 ± 0.004	0.69	2.77
$L(\phi,\theta,\text{MFCC})$	4.52 ± 0.003	0.70	2.79
$L(\phi,\theta,\text{STFT},\text{MFCC})$	4.53 ± 0.002	0.69	2.76
$L(\phi,\theta,\text{SC}^2,\text{MAG}^2)$	4.53 ± 0.002	0.69	2.78

5. Conclusion

In this work, we integrated metric based regularizers into the StoRM architecture, leading to significant improvement in training convergence as in Figure. 3 and slight improvements in dereverberation as in Table 1. Additionally, T-biased sampling shows even faster convergence with nearly similar performance, whereas 0-biased sampling degrades performance, as in Figure 4 and Table 1. We also investigated the optimality of perceptual and recognition performance of metric-guided StoRM compared to the baseline. Metric guidance ensured quality-aware dereverberation, with the trained model acting as nearly identity transformation for clean speech w.r.t baseline where PESQ drops (4.5 → 4.24) as evident from Tables 2 & 3 jointly.

6. Acknowledgement

The authors would like to express their sincere gratitude to the Centre for Artificial Intelligence and Robotics (CAIR), Defence Research and Development Organisation (DRDO), Government of India, for their support and encouragement throughout this work. Their guidance and resources were instrumental in the successful completion of this work.

7. References

[1] J. Ma, W. Wang, Y. Yang, and F. Zheng, "Mutual learning for acoustic matching and dereverberation via visual scene-driven diffusion," *arXiv preprint arXiv:2407.10373*, 2024.

[2] S. Arora, A. Risteski, and Y. Zhang, "Theoretical limitations of encoder-decoder gan architectures," *arXiv preprint arXiv:1711.02651*, 2017.

[3] A. V. Oppenheim and R. W. Schafer, "Digital signal processing prentice-hall," *Englewood Cliffs, NJ*, vol. 19752, pp. 26–30, 1975.

[4] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 2, pp. 113–120, 1979.

[5] B. Widrow and S. D. Stearns, *Adaptive signal processing*. USA: Prentice-Hall, Inc., 1985.

[6] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning (ICML)*, 2008, pp. 1096–1103.

[7] A. Défossez, G. Synnaeve, and Y. Adi, "Real time speech enhancement in the waveform domain," in *Interspeech*, 2020, pp. 3291–3295.

[8] H. Fang, G. Carbajal, S. Wermter, and T. Gerkmann, "Variational autoencoder for speech enhancement with a noise-aware encoder," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 676–680.

[9] S. Abdulatif, R. Cao, and B. Yang, "CMGAN: Conformer-based metric-gan for monaural speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 2477–2493, 2024.

[10] H. Liu, X. Liu, Q. Kong, Q. Tian, Y. Zhao, D. Wang, C. Huang, and Y. Wang, "Voicefixer: A unified framework for high-fidelity speech restoration," in *Interspeech*, 2022, pp. 4232–4236.

[11] J. Su, Z. Jin, and A. Finkelstein, "Hifi-gan: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks," in *Interspeech*, 2020, pp. 4506–4510.

[12] S. Welker, J. Richter, and T. Gerkmann, "Speech enhancement with score-based generative models in the complex stft domain," in *Interspeech*, 2022, pp. 2928–2932.

[13] J.-M. Lemercier, J. Richter, S. Welker, and T. Gerkmann, "Storm: A diffusion-based stochastic regeneration model for speech enhancement and dereverberation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.

[14] J.-M. Lemercier, E. Moliner, S. Welker, V. Välimäki, and T. Gerkmann, "Unsupervised blind joint dereverberation and room acoustics estimation with diffusion models," *arXiv preprint arXiv:2408.07472*, 2024.

[15] R. Yang, Z. Wang, B. Jiang, and S. Li, "The convergence of variance exploding diffusion models under the manifold hypothesis," *Submitted to International Conference on Learning Representations (ICLR), Feb. arXiv preprint arXiv:2309.11645*, 2024.

[16] N. Ikeda and S. Watanabe, *Stochastic Differential Equations and Diffusion Processes*, ser. North-Holland Mathematical Library. North Holland, 2014. [Online]. Available: https://books.google.co.in/books?id=QZbOBQAAQBAJ

[17] D. B. Paul and J. M. Baker, "The design for the wall street journal-based CSR corpus," in *Proceedings of the Workshop on Speech and Natural Language*, ser. HLT '91. USA: Association for Computational Linguistics, 1992, p. 357–362.

[18] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[19] S.-W. Fu, C. Yu, K.-H. Hung, M. Ravanelli, and Y. Tsao, "Metricgan-u: Unsupervised speech enhancement/dereverberation based only on noisy/reverberated speech," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7412–7416.

[20] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, J.-C. Chou, S.-L. Yeh, S.-W. Fu, C.-F. Liao, E. Rastorgueva, F. Grondin, W. Aris, H. Na, Y. Gao, R. D. Mori, and Y. Bengio, "SpeechBrain: A general-purpose speech toolkit," 2021, arXiv:2106.04624.

[21] M. Ravanelli, T. Parcollet, A. Moumen, S. De Langen, C. Subakan, P. Plantinga, Y. Wang, P. Mousavi, L. D. Libera, A. Ploujnikov, F. Paissan, D. Borra, S. Zaiem, Z. Zhao, S. Zhang, G. Karakasidis, S.-L. Yeh, P. Champion, A. Rouhe, R. Braun, F. Mai, J. Zuluaga-Gomez, S. M. Mousavi, A. Nautsch, H. Nguyen, X. Liu, S. Sagar, J. Duret, S. Mdhaffar, G. Laperrière, M. Rouvier, R. De Mori, and Y. Estève, "Open-source conversational ai with speechbrain 1.0," *J. Mach. Learn. Res.*, vol. 25, no. 1, Jan. 2024.

[22] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," *arXiv preprint arXiv:1904.05862*, 2019.

[23] T. R. Letowski and A. A. Scharine, "Correlational analysis of speech intelligibility tests and metrics for speech transmission," 2017. [Online]. Available: http://rgdoi.net/10.13140/RG.2.2.26581.93921

[24] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Proceedings (Cat. No. 01CH37221*, vol. 2. IEEE, 2001, pp. 749–752.

[25] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2010, pp. 4214–4217.

[26] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, 2007.

[27] ——, "Evaluation of objective measures for speech enhancement," in *Interspeech*, 2006.